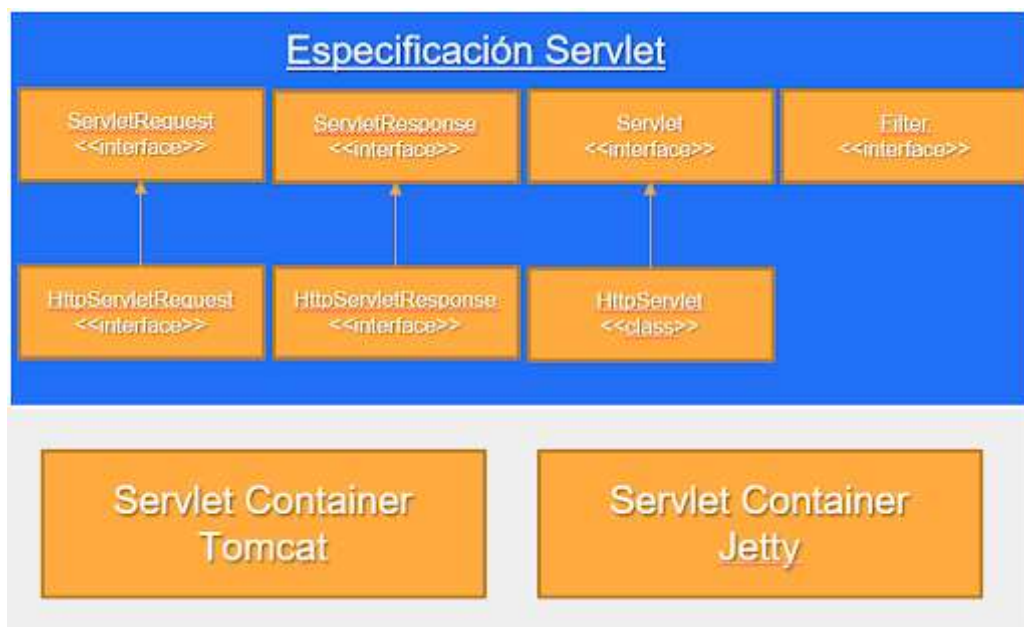


## Para saber más: Servlet x HttpServlet

Hablamos sobre la especificación Servlet y la garantía de que podemos ejecutar nuestro proyecto en diferentes Servlet Containers. Presenté la imagen a continuación, que muestra algunos tipos importantes de la especificación, como `jakarta.servlet.Servlet`, `jakarta.servlet.http.HttpServlet`, `jakarta.servlet.ServletRequest`, `jakarta.servlet.http.HttpServletRequest`, entre otros:



Tenga en cuenta que hay tipos más genéricos y tipos más específicos, centrados en el protocolo HTTP, por ejemplo:

- `jakarta.servlet.Servlet` → `jakarta.servlet.http.HttpServlet`
- `jakarta.servlet.ServletRequest` →  
`jakarta.servlet.http.HttpServletRequest`
- `jakarta.servlet.ServletResponse` →  
`jakarta.servlet.http.HttpServletResponse`

Además de la interface `Filter` , siempre hay un tipo más específico del mundo HTTP. ¿Por qué?

La idea inicial era que el mundo Servlets soportara otros protocolos como FTP o SMTP. Es decir, los servlets y los servlet containers podrían funcionar con otros protocolos además del HTTP. Es por eso que existen estas interfaces genéricas (sin `Http` en el nombre), para extender y servir a nuevos protocolos.

Por ejemplo, podría haber un FTP Servlet Container, que admitiera el protocolo FTP y, por lo tanto, ampliaría los tipos genéricos para crear un `FtpServlet` o un `FtpServletRequest` . Al final, no existen tales implementaciones, y el protocolo HTTP es el único que admite servlets. Esto también se debe a la omnipresencia del protocolo HTTP en el día a día, y la poca relevancia de otros protocolos.